

**PROGRAM SHARING METHOD USING BOARD IDENTIFICATIONS IN A  
MOBILE COMMUNICATION SYSTEM**

**PRIORITY**

5           This application claims priority to an application entitled "Program Sharing Method Using Board IDs in Mobile Communication System" filed in the Korean Industrial Property Office on February 6, 2001 and assigned Serial No. 2001-5578, the contents of which are hereby incorporated by reference.

**BACKGROUND OF THE INVENTION**

**1. Field of the Invention**

10           The present invention relates generally to a mobile communication system, and in particular, to a method for sharing an execution file among a plurality of target boards using board IDs in a mobile communication system.

15           **2. Description of the Related Art**

As shown in FIG. 1, a mobile communication system generally includes an MSC (Mobile Switching Center) 100, a BSM (Base Station Manager) 106, BSSs (Base Station Systems) 102 and 104, BTSs (Base Transceiver Stations) 108, 110, and 112, and MSs (Mobile Stations) (not shown). BSS 102 is connected to a plurality of BTSs 108, 110, and 112, for transmitting signals received from BTSs 108, 110, and 112 to MSC 100 and transmitting signals received from MSC 100 to MSs through BTSs 108, 110, and 112. BSS 102 is provided with target function boards to process signals exchanged between

MSC 100 and BTSs 108, 110, and 112. Though not shown in FIG. 1, BSS 104 would have a similar configuration as to that of BSS 102, including the target function boards.

Execution files are prepared separately for respective target boards in a BSS as shown in Table 1 and managed by the BSM.

5

Table 1

Target Board		A	B	C	...	n
Execution file	Separate loading & running	Boot, OS, AP for target A	Boot, OS, AP for target B	Boot, OS, AP for target C	...	Boot, OS, AP for target n
	Integrated loading & running	Execution file for target A	Execution file for target B	Execution file for target C	...	Execution file for target n

In the conventional mobile communication system a plurality of target boards (Target Board A, B, C...n) are included in each BSS. Each target board requires its own respective execution files. Each target board must store and execute the files within its own resources. This results in duplication of program files stored in the same BSS, and results in management complexity for the BSS in the BSM.

### SUMMARY OF THE INVENTION

15 It is, therefore, an object of the present invention to provide a method of sharing an execution file among a plurality of target boards using board identifications (IDs) in a mobile communication system.

The foregoing and other objects can be achieved by providing a program sharing method using board IDs in a mobile communication system. When power to the system

is turned on, a shared execution file is executed for a plurality of target boards in the system. The ID of each target board is read and hardware is initialized according to the target board IDs. An operating system (OS) for each target board is initialized according to the ID of the target board. A branch routine for each target board is invoked according to the target board ID, and an application program is implemented for the target board according to the target board ID.

### BRIEF DESCRIPTION OF THE DRAWINGS

The above and other objects, features and advantages of the present invention will become more apparent from the following detailed description when taken in conjunction with the accompanying drawings in which:

FIG. 1 illustrates the configuration of a mobile communication system to which an embodiment of the present invention is applicable;

FIG. 2 is a schematic view illustrating a BSM system to which the embodiment of the present invention is applicable; and

FIG. 3 is a flowchart illustrating a program sharing method using board IDs according to the embodiment of the present invention.

### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

A preferred embodiment of the present invention will be described hereinbelow with reference to the accompanying drawings. In the following description, well-known functions or constructions are not described in detail since they would obscure the

invention in unnecessary detail.

FIG. 2 illustrates a BSM system according to an embodiment of the present invention. Referring to FIG. 2, BSM 106 stores an execution file that is to be shared among a plurality of target boards 200, 202, and 204 in a BSS 102. This configuration is shown in Table 2. BSM 106 implements the execution file for each board when power is initially supplied to the system.

Table 2

Target board		A	B	C	...	n
Execution file	Separate loading & running	One shared boot, OS, and AP				
	Integrated loading & running	One shared execution file				

As shown in Table 2, BSM 106 manages a shared execution file for target boards 200, 202, and 204 located in BSS 102 and assigns unique IDs to target boards 200, 202 and 204 according to the embodiment of the present invention. The shared execution file is compiled as a single execution file to include program code for each target board. Upon the system running the execution file, the program branches into corresponding routines for different target board functions according to the target board IDs. The target board IDs are assigned in both software and hardware so that they cannot be deleted at power off.

FIG. 3 is a flowchart illustrating a control operation of a BSM for sharing a single execution file among a plurality of target boards of a BSS according to the embodiment of the present invention.

Referring to FIGs. 2 and 3, in step 300 it is determined if all target boards 200, 202, and 204 in BSS 102 are powered-on. If it is determined that not all target boards 200, 202 and 204 are powered-on the process returns to start. This insures that all of the target boards are in a state for the execution file to be run. If it is determined in step 300 that all of the target boards 200, 202 and 204 are powered-up, then in step 302 BSM 106 executes a shared execution file for the target boards 200, 202, and 204. The processor of each target board reads a corresponding board ID and sets an arbitrary variable to the board ID in a predetermined area of a memory in step 304. The processor initializes the hardware according to the target ID in step 306. The hardware initialization varies with the hardware structure in each target board.

Next in step 308, the processor initializes an operating system (OS) for the corresponding target board using the target board ID. Since the OS also varies according to the hardware structure or running conditions, the OS operates differently according to target board IDs. The processor branches into processing sub-routines for the corresponding target board according to the target board ID and executes a corresponding application program in step 310. The application program may be different according to target board functions. Therefore, the processor reads the corresponding target board ID and branches into a corresponding application routine. Consequently, each target board can be operated with a single shared execution file for a common function in the target boards.

In accordance with the present invention as described above, a single shared execution file is stored and executed for a plurality of target boards using target board

IDs in a mobile communication system. As a result, multiple execution files are integrated into one master execution file, thus reducing the memory requirements of the overall system.

While the invention has been shown and described with reference to a certain preferred embodiment thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined by the appended claims.

10007059-10007059